AMENDMENTS TO THE CLAIMS

Please cancel Claims 45-51, 53, 60-66, 68-70, 72, and 76 without prejudice, as indicated below.

Please add Claims 77-97, as indicated below.

Please amend Claims 41, 54, 71, and 73-75 as indicated below.

A complete listing of all claims is presented below with insertions underlined (e.g., insertion), and deletions struckthrough or in double brackets (e.g., ~~deletion~~ or [[deletion]]):

1.-40. (Canceled)

41. (Currently Amended) A method for defining a system specification for a heterogeneous digital system, said method comprising the steps of:

partitioning said heterogeneous digital system into a plurality of processes, each of the processes having a defined behavior and each of the processes having at least one control thread;

defining separately from said processes a single data independent data communication protocol for communication within said heterogeneous digital system and between said processes;

configuring data communication interfaces in the form of communication input ports and communication output ports for each of the processes, the communication ports forming memory free communication channels; and

combining the results of the steps of partitioning, defining and configuring to define specifications for said plurality of processes to form said system specification.

42. (Previously Presented) The method of Claim 41, wherein the specification for a first process is independent of the specification of a second process.

43. (Previously Presented) The method of Claim 41, further comprising the step of duplicating the specification for a process of a first system for a process of a second system.

44. (Previously Presented) The method of Claim 41, wherein said step of configuring data communication interfaces involves defining communication interfaces with input ports and output ports to provide unidirectional, point-to-point connections between input ports of a first process and output ports of a second process, said input ports and said output ports being part of the associated processes.

45.-51. (Cancelled)

52. (Previously Presented) The method of Claim 44, wherein said processes are implemented in a hardware description language or in a programming language.

53. (Cancelled)

54. (Currently Amended) A method of implementing a heterogeneous digital system comprising the steps of:

partitioning said heterogeneous digital system into a plurality of processes, each process having a defined behavior and with at least one control thread;

defining separately from said processes, a single data independent data communication protocol for communication within said heterogeneous digital system and between said processes;

organizing said data communication protocol with input and output ports for said processes, said ports using memory free communication channels; and

designing a plurality of processors to implement said process.

55. (Previously Presented) The method of Claim 54, wherein said step of designing processors comprises the step of specifying a processor having specification which conform to the processes implemented.

56. (Previously Presented) The method of Claim 55, wherein said processor comprises a programmable, general purpose processor.

57. (Previously Presented) The method of Claim 55, wherein said processor comprises a programmable digital signal processor.

58. (Previously Presented) The method of Claim 55, wherein said processor comprises a dedicated, custom processor.

59. (Previously Presented) The method of Claim 55, wherein said processor comprises custom logic circuitry with a controller such that the resulting digital system operates according to functional and real-time specifications.

60.-66. (Cancelled)

67. (Previously Presented) The method of Claim 55, wherein said communication ports connect processes defined of at least one of a plurality of specifications.

68.-70. (Cancelled)

71. (Currently Amended) The method recited in Claim [[56]]54, wherein said communication channels are implemented in integrated circuit form for communications between a

first processor and a second processor across said channel, said first and second processors selected from one or more of a plurality of processor types.

72. (Cancelled)

73. (Currently Amended) The method recited in Claim [[56]]54, wherein said communication channels are implemented in software for communications between a first processor and a second processor across said channel, said first and second processors selected from one or more of a plurality of processor types.

74. (Currently Amended) The method recited in Claim [[56]]54, wherein said communication channels are implemented in a combination of hardware and software, for communications between a first processor and a second processor across said channel, said first and second processors selected from one or more of a plurality of processor types.

75. (Currently Amended) The method recited in Claim [[57]]54, wherein said step of partitioning involves defining a library of auxiliary processes to simulate the heterogeneous digital system, the library of processes selected from a plurality of processes.

76. (Cancelled)

77. (New) A method for defining a system specification for an integrated circuit, said method comprising:

partitioning said integrated circuit into a plurality of processes, each of the processes having a defined behavior and each of the processes having at least one control thread;

defining separately from said processes a single data independent data communication protocol for communication within said integrated circuit and between said processes;

configuring data communication interfaces in the form of communication input ports and communication output ports for each of the processes of said integrated circuit, the communication ports forming memory free communication channels between said processes of said integrated circuit; and

combining the results of the steps of partitioning, defining and configuring to define specifications for said plurality of processes to form said system specification.

78. (New) The method of Claim 77, wherein the specification for a first process is independent of the specification of a second process.

79. (New) The method of Claim 77, further comprising the step of duplicating the specification for a process of a first system for a process of a second system.

80. (New) The method of Claim 77, wherein said step of configuring data communication interfaces involves defining communication interfaces with input ports and output ports to provide unidirectional, point-to-point connections between input ports of a first process and output ports of a second process, said input ports and said output ports being part of the associated processes.

81. (New) The method of Claim 80, wherein said processes are implemented in a hardware description language or in a programming language.

82. (New) A method of implementing an integrated circuit, said method comprising:

partitioning said integrated circuit into a plurality of processes, each process having a defined behavior and with at least one control thread;

defining separately from said processes, a single data independent data communication protocol for communication within said integrated circuit and between said processes of said integrated circuit;

organizing said data communication protocol with input and output ports for said processes, said ports using memory free communication channels; and

designing a plurality of processors to implement said process.

83. (New) The method of Claim 82, wherein said step of designing processors comprises the step of specifying a processor having specification which conform to the processes implemented.

84. (New) The method of Claim 83, wherein said processor comprises a programmable, general purpose processor.

85. (New) The method of Claim 83, wherein said processor comprises a programmable digital signal processor.

86. (New) The method of Claim 83, wherein said processor comprises a dedicated, custom processor.

87. (New) The method of Claim 83, wherein said processor comprises custom logic circuitry with a controller such that the resulting digital system operates according to functional and real-time specifications.

88. (New) The method of Claim 83, wherein said communication ports connect processes defined of at least one of a plurality of specifications.

89. (New) The method recited in Claim 82, wherein said communication channels are implemented in integrated circuit form for communications between a first processor and a second processor across said channel, said first and second processors selected from one or more of a plurality of processor types.

90. (New) The method recited in Claim 82, wherein said communication channels are implemented in software for communications between a first processor and a second processor across said channel, said first and second processors selected from one or more of a plurality of processor types.

91. (New) The method recited in Claim 82, wherein said communication channels are implemented in a combination of hardware and software, for communications between a first processor and a second processor across said channel, said first and second processors selected from one or more of a plurality of processor types.

92. (New) The method recited in Claim 82, wherein said step of partitioning involves defining a library of auxiliary processes to simulate the integrated circuit, the library of processes selected from a plurality of processes.

93. (New) A method for defining a system specification for a digital system, said method comprising the steps of:

partitioning said system into a plurality of processes, each of the processes having a defined behavior and each of the processes having at least one control thread;

defining separately from said processes a single data independent data communication protocol for communication within said digital system and between said processes;

configuring data communication interfaces in the form of communication input ports and communication output ports for each of the processes, the communication ports forming memory free communication channels, said step of configuring data communication interfaces involving defining communication interfaces with input ports of a first process and output ports to provide unidirectional, point-to-point connections between input ports of a first process and output ports of a second process, said input

ports and said output ports being part of the associated processes, said processes implemented in C, Silage or VHDL language; and

combining the results of the steps of partitioning, defining and configuring to define specifications for said plurality of processes to form said system specification.

94. (New) A method of implementing a digital system comprising the steps of:

partitioning said system into a plurality of processes, each process having a defined behavior and with at least one control thread;

defining separately from said processes, a single data independent data communication protocol for communication within said digital system and between said processes;

organizing said data communication protocol with input and output ports for said processes, said ports using memory free communication channels; and

designing a plurality of processors to implement said process, said step of designing processors comprising the step of specifying a processor having specification which conforms to the processes implemented, said processor comprising a programmable digital signal processor, wherein said plurality of specifications are selected from a group consisting of Silage descriptions, C descriptions, VHDL process descriptions.

95. (New) A method of implementing a digital system comprising the steps of:

partitioning said system into a plurality of processes, each process having a defined behavior and with at least one control thread;

defining separately from said processes, a single data independent data communication protocol for communication within said digital system and between said processes;

organizing said data communication protocol with input and output ports for said processes, said ports using memory free communication channels implemented as interrupt driven I/O; and

designing a plurality of processors to implement said process, said step of designing processors comprising the step of specifying a processor having specification which conforms to the processes implemented.

96. (New) A method of implementing a digital system comprising the steps of:

partitioning said system into a plurality of processes, each process having a defined behavior and with at least one control thread;

defining separately from said processes, a single data independent data communication protocol for communication within said digital system and between said processes;

organizing said data communication protocol with input and output ports for said processes, said ports using memory free communication channels, said communication channels implemented in integrated circuit form for communications between a first processor and a second processor across said channel, said first and second processors selected from one or more of a plurality of processor types, wherein said plurality of processor types consists of Cathedral-III processors, ARM processors and VHDL generated processors; and

designing a plurality of processors to implement said process, said step of designing processors comprising the step of specifying a processor having specification which conforms to the processes implemented, said processor comprising a programmable, general purpose processor.

97. (New) A method of implementing a digital system comprising the steps of:

partitioning said system into a plurality of processes, each process having a defined behavior and with at least one control thread, said step of partitioning involving defining a library of auxiliary processes to simulate the digital system, the library of processes selected from a plurality of processes consisting of one or more of an interactive I/O process, a file I/O process, a graphical output process, a channel duplicator process, a channel merging process, a FF process, a slider process, a button process, a first-in, first-out buffer process, an ARM processor, a digital to analog conversion process and an analog to digital conversion process;

defining separately from said processes, a single data independent data communication protocol for communication within said digital system and between said processes;

organizing said data communication protocol with input and output ports for said processes, said ports using memory free communication channels; and

designing a plurality of processors to implement said process, said step of designing processors comprising the step of specifying a processor having specification which conforms to the processes implemented, wherein said processor comprises a programmable digital signal processor.